Knowledge-Based Systems 76 (2015) 67–78

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Support vector machine-based optimized decision threshold adjustment strategy for classifying imbalanced data

Hualong Yu [a,b,c], Chaoxu Mu [a,b], Changyin Sun [a,b,*], Wankou Yang [a,b], Xibei Yang [c], Xin Zuo [c]

[a] School of Automation, Southeast University, Nanjing 210096, China
[b] Key Lab of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Southeast University, Nanjing 210096, China
[c] School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China

ARTICLE INFO

Article history:
Received 3 May 2014
Received in revised form 4 December 2014
Accepted 5 December 2014
Available online 13 December 2014

Keywords:
Class imbalance
Support vector machine
Decision threshold adjustment
Optimization search
Ensemble learning

ABSTRACT

Class imbalance problem occurs when the number of training instances belonging to different classes are clearly different. In this scenario, many traditional classifiers often fail to provide excellent enough classification performance, i.e., the accuracy of the majority class is usually much higher than that of the minority class. In this article, we consider to deal with class imbalance problem by utilizing support vector machine (SVM) classifier with an optimized decision threshold adjustment strategy (SVM-OTHR), which answers a puzzled question: how far the classification hyperplane should be moved towards the majority class? Specifically, the proposed strategy is self-adapting and can find the optimal moving distance of the classification hyperplane according to the real distributions of training samples. Furthermore, we also extend the strategy to develop an ensemble version (EnSVM-OTHR) that can further improve the classification performance. Two proposed algorithms are both compared with many state-of-the-art classifiers on 30 skewed data sets acquired from Keel data set Repository by using two popular class imbalance evaluation metrics: F-measure and G-mean. The statistical results of the experiments indicate their superiority.

1. Introduction

In the past decade, the class imbalance problem has received considerable attention in several fields, such as artificial intelligence [1], machine learning [2] and data mining [3,4]. A data set is said to be imbalanced when and only when the instances of some classes are obviously much more than that in other classes. The problem is important due to it widely emerges in many real-world applications, including financial fraud detection [5], network intrusion detection [6], spam filtering [7], video monitoring [8], medical diagnosis [9], Bioinformatics [10], etc. Generally, in these applications, we are more interested in the pattern represented by the examples of the minority class. However, majority traditional classification algorithms pursuing the minimal training errors would heavily damage the recognition accuracy of the minority class, thus it is necessary to adopt some bias correction techniques before/after constructing a classifier.

The bias correction techniques can be roughly divided into four major categories as follows:

1. Resampling the original training set until all the classes are approximately equally represented. Resampling includes oversampling [11–13], undersampling [14,15] and hybrid sampling [16].
2. Cost-sensitive learning, which is also called instances weighting method, assigns different weights for the training instances belonging to different classes so that the misclassification of the minority class can be highlighted [17–19].
3. Moving the decision boundary (decision threshold adjustment) towards the majority class in order to remedy the bias caused by skewed sample distributions [20,21]. Unlike the other correction techniques, decision threshold adjustment strategy runs after modeling a classifier.
4. Ensemble learning that provides a framework to incorporate resampling strategy, weighting strategy or decision threshold adjustment strategy, usually produces better and more balanced classification performance [22–29].

Among those correction techniques mentioned above, decision threshold adjustment is regarded as a potential solution for dealing with class imbalance in recent studies [20,21]. However, the existing decision threshold adjustment approaches generally give the moving distance of classification boundary empirically, thus fail

* Corresponding author at: School of Automation, Southeast University, No. 2 Sipailou, Nanjing 210096, China. Tel./fax: +86 25 83794974.
E-mail address: cysun@seu.edu.cn (C. Sun).

http://dx.doi.org/10.1016/j.knosys.2014.12.007
0950-7051/

to answer a significant question: how far the classification hyperplane should be moved towards the majority class? This study solves this puzzle in the context of support vector machine (SVM) [30]. SVM is a robust classifier and is relatively insensitive to class imbalance in comparison with many other classification algorithms, because its classification hyperplane only associates with a few support vectors [31].

In this paper, we first investigate the reason that the classification performance of SVM can be destroyed by skewed classification data in theory, and then we analyze the merits and drawbacks of some existing SVM-based bias correction techniques. Next, the computational formula of the moving distance in SVM-THR algorithm proposed by Lin and Chen [21] is intensively modified to lead to one optimized version (SVM-OTHR). Furthermore, we incorporate SVM-OTHR into Bagging ensemble learning framework and present a novel classification algorithm named EnSVM-OTHR. In particular, to avoid overfitting and to guarantee the diversity of different individuals, a small random perturbation term is inserted into each SVM-OTHR to disturb the final position of classification hyperplane. Finally, we compare the two proposed classification algorithms with many state-of-the-art imbalanced classifiers on 30 data sets acquired from Keel data set Repository via non-parametrical statistical testing [32,33], indicating their superiority.

The rest of this paper is organized as follows. In Section 2, we introduce SVM theory and explain the reason that the performance of SVM can be damaged by imbalanced classification data. Section 3 briefly reviews some existing SVM-based class imbalance correction techniques and indicates their pros and cons. In Section 4, one optimized SVM decision threshold adjustment strategy (SVM-OTHR) and its extended version based on ensemble learning (EnSVM-OTHR) are described in detail. Experimental results and discussions are presented in Section 5. Finally in Section 6, the main contributions of this study are summarized.

2. Why SVM can be damaged by class imbalance ?

Support vector machine (SVM), which comes out of the theory of structure risk minimization, has several merits as follows: high generalization capability, absence of local minima and adaptation for high-dimension and small sample data [31,34].

Given some training data D, a set of m points of the form: $D = \{(x_i, y_i)|x_i \in \mathbb{R}^p, y_i \in \{-1, 1\}_{i=1}^m\}$, where y_i is either 1 or -1, indicating the class to which the point x_i belongs. Each x_i is one p-dimensional real vector. SVM is used to find the maximum margin hyperplane that divides the points having $y_i = 1$ from those having $y_i = -1$. The decision function of SVM is described as:

$$h(x) = \langle w, \phi(x)\rangle + b \tag{1}$$

where $\phi(x)$ represents a mapping of sample x from the input space to high-dimensional feature space, $\langle\cdot,\cdot\rangle$ denotes the dot product in the feature space, w denotes the weight vector for learned decision hyperplane and b is the model bias. We can optimize the values of w and b by solving the following optimization problem:

$$\begin{aligned} &\text{minimize: } g(w,\xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \\ &\text{subject to: } y_i(\langle w, \phi(x_i)\rangle + b) \geqslant 1 - \xi_i, \quad \xi_i \geqslant 0 \end{aligned} \tag{2}$$

where ξ_i is the ith slack variable and C is regularization parameter (penalty factor) which is used to regulate the relationship between training accuracy and generalization. Then the minimization problem in formula (2) can be transformed to a dual form and be rewritten as:

$$\begin{aligned} &\text{maximize: } W(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \\ &\text{subject to: } \sum_{i=1}^{m} y_i \alpha_i = 0, \quad \forall i: \ 0 \leqslant \alpha_i \leqslant C \end{aligned} \tag{3}$$

where α_i is the sample x_i's lagrange multiplier, $K(\cdot,\cdot)$ is a kernel function that maps the input vectors into a suitable feature space:

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j)\rangle \tag{4}$$

Some previous work has found that radial basis kernel function (RBF) generally provides better classification accuracy than many other kernel functions [31,34]. RBF kernel is presented as follows:

$$K(x_i, x_j) = \exp\left\{-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right\} \tag{5}$$

where σ is the width of RBF kernel.

Although previous work found that SVM is more robust to class imbalance than many other machine learning methods as its classification hyperplane only associates with a few support vectors, it can be still hurt by skewed class distributions to some extent. We try to analyze its reason in theory.

After training an SVM classifier, lagrange multiplier α_i can be divided into three categories as follows:

Case 1: $\alpha_i = 0$, it means the instance x_i is classified accurately.
Case 2: $0 < \alpha_i < C$, the corresponding instance x_i is called a normal support vector which is exactly on one of the margin hyperplanes.
Case 3: $\alpha_i = C$, x_i is called a boundary support vector that lies between margins. The percentage of boundary support vectors reflects the error rate of SVM to some extent.

Suppose N^+ and N^- represent the number of instances belonging to the positive class (minority class) and the negative class (majority class), respectively. N_{sv}^+ and N_{sv}^- are the number of support vectors (including normal and boundary support vectors) in two classes, while N_{boundary}^+ and N_{boundary}^- represent the number of boundary support vectors in two classes, respectively. According to formula (3), we can get:

$$\sum_{i=1}^{m}\alpha_i = \sum_{y_i=+1}\alpha_i + \sum_{y_i=-1}\alpha_i \tag{6}$$

$$\sum_{y_i=+1}\alpha_i = \sum_{y_i=-1}\alpha_i \tag{7}$$

Because α_i's value is C at most, it can deduce two inequalities as follows:

$$\sum_{y_i=+1}\alpha_i \geqslant N_{\text{boundary}}^+ \times C \tag{8}$$

$$\sum_{y_i=+1}\alpha_i \leqslant N_{sv}^+ \times C \tag{9}$$

By integrating formula (8) and (9), we get:

$$N_{sv}^+ \times C \geqslant \sum_{y_i=+1}\alpha_i \geqslant N_{\text{boundary}}^+ \times C \tag{10}$$

Similarly, it is not difficult to get the following inequality:

$$N_{sv}^- \times C \geqslant \sum_{y_i=-1}\alpha_i \geqslant N_{\text{boundary}}^- \times C \tag{11}$$

Suppose $\sum_{y_i=+1}\alpha_i = \sum_{y_i=-1}\alpha_i = M$, if formula (10) and (11) respectively divide by $N^+ \times C$ and $N^- \times C$, we get: